Generation and Use of a Digest System by Integrating OCR and Smart Searches

Germán Cáseres[1], Lisandro Delía[1], Pablo Thomas[1], Verónica Aguirre[1],

[1] Instituto de Investigación en Informática III-LIDI, Universidad Nacional de La Plata, La Plata, Buenos Aires, Argentina
{gcaseres, ldelia, pthomas, vaguirre}@lidi.info.unlp.edu.ar

Abstract. A digest can be defined as a regulations repository which is manipulated by organizations for extended time periods. The search for information in this repositories can be tedious without assistance from an ad-hoc software application. This work presents the development of a Digest Software System with its architecture and integration with other base tools. Finally, two study cases are presented where the developed product is used.

Keywords: digest, full text search, ocr, solr, tika

1 Introduction

Organizations usually have a vast set of pre-established regulations that have to be consulted regularly. Oftentimes, applicable laws defined in those standards are not known and searches must be done that result in tedious document reviews.

Clearly, it would be convenient to have some sort of repository where the entire legislation is stored. However, searching the regulatory framework in relation to any given issue may involve some time to collect and analyze current regulations.

In this context, there is a clear need to have a digest, i.e., a collection of all regulations applicable to a specific field. This digest should also be digital in nature, since it would not have a significant physical fingerprint, but it should also be integrated to a system that provides a certain level of intelligence to search for ordinances or resolutions on a specific topic.

Thus, there are two essential processes in a software system that acts as an institutional digest. First, regulations have to be digitized, considering that many of those might be available in printed format only, and which will therefore have to be converted to a digital format. Naturally, any dispositions or regulations issued today by institutions are already produced in digital format. All this constitutes a digital repository or database of regulations.

Secondly, such repository should include a feature to perform “smart” searches that consider the entire text of the regulations, showing results sorted by relevance (word similarity, proximity, and so forth). These functionalities are available in current popular web search engines.

In this article, we introduce the development and use of a software system that is built as an institutional digest. The rest of this paper is organized as follows: in

Section 2, the problem at hand is introduced. Then, in Section 3, the tools selected to solve the problem are described. Sections 4 and 5 deal with the digest system Digesto and its use. After that, study cases and the results obtained are detailed. Finally, conclusions and future lines of work are discussed.

2 Description of the problem

The digitization process of a standard that is on paper support does not involve just obtaining a digital representation of its contents, but it also requires a mechanism that can interpret the textual information present in the digitized document. In general, this mechanism is not simple and requires applying image recognition algorithms capable of identifying language symbols.

Once this information is obtained, a mechanism that not only allows storing the text content of the regulations, but also ensures a fast response time for arbitrary content searches, has to be selected.

To achieve this, information search indexes must be generated based on the syntactic and semantic characteristics of the language used in the standards.

These mechanisms should be also integrated in a single system that can perform the necessary processes transparently to the user in charge of loading up the regulations or performing searches on them.

2.1 Digitization of Regulations

The digitization of a regulation is done using a device that is capable of capturing the content printed on the paper and saving it as a digital image. To do this task, a scanner or a digital photographic camera can be used.

Once a digital format image of the regulation is obtained, a recognition process has to be run to extract the text information that will be used as foundation for the searches. This process, called OCR (Optical Character Recognition) [1], emulates the ability of the human eye to recognize objects, identifying the symbols in an image that correspond to those of a specific alphabet.

Images enter the OCR process in a digital format (pixel matrix) and go through a number of stages that apply transformations on them to simplify the task of identifying the symbols in different contexts of colors, locations, designs, etc.

Process output information consists of the symbols extracted from the image in some character coding format (ASCII, UTF8, etc.). This output format is the most suitable to facilitate text processing and storage in any computer system.

The quality of the result depends on the quality of the digitization process and the characteristics of the algorithms used to carry out the process.

2.2 Storing Information for Content Searches

Storing the textual content of a regulation is a task that could be solved by simply storing the text as an attribute of a relational database table. However, solving this following such a simple method would involve leaving out search "smart" features, such as the ones that can be found in the most popular search engines.

To store text that is ready to allow quick searches and yields relevant results, different indexing techniques are required, which are usually not developed in the relational database engines [2] that are currently used (MySQL, PostgreSQL, SQL Server, and so forth).

3 Selecting the Right Tools

To solve the problems described above, specific knowledge is required in the field of image recognition as well as in relation to storing and indexing large volumes of information. For this reason, we decided to select tools that can solve these issues and can also be configured and integrated into a system at a specific domain, such as the Digesto system.

To perform the tasks related to text recognition on images, the TesseractOCR library [3] was used, which is an open source optical character recognition engine whose development began in 1995 and which has been optimized and improved by the Google organization since 2006 and up to the present time.

In the case of the search engine, an existing technology called Apache Solr [4] was selected; this technology makes the development, configuration and utilization of the various information indexing techniques easier, while providing a dynamic querying mechanism and ease of integration with other systems.

Solr is an open source search platform built on the Apache Lucene [5] project.

Its main features are:

- Advanced text search capabilities (through Lucene)
- Optimization for large traffic volumes
- Based on standards (XML, JSON and HTTP) [6]
- Scalability and fault-tolerance
- Flexibilization and adaptability
- Extensible architecture through plug-ins

Solr uses the Lucene library to carry out indexation and text querying tasks. At the same time, it also presents an adaptable and extensible platform that can manage the different indexing or querying requirements, without affecting system stability.

Indexed information storage is also managed by Solr through its own implementation of a non-SQL database [7], which allows easily and quickly adapting to any data structure.

The use of standards such as XML, JSON and HTTP allows simple integration with other systems, since indexing the contents of a regulation or obtaining a set of results based on a search criterion simply involves running an HTTP requirement against the Solr server.

The plug-in extensible architecture allows integrating other components to extend tool capabilities. In this sense, one of the plug-ins available in the platform belongs to the same organization that develops Solr, and is called Apache Tika [8].

Tika is a plug-in that is independent from Solr and whose purpose is obtaining information from different popular file formats:

- Images (gif, png, bmp, etc.)
- Office packages (xls, doc, ppt, etc.)
- Compression and packing (zip, tar, gz, etc.)
- Structured text (xml, html, xaml, etc.)
- Portable document format (.pdf)
- Etc.

With Tika, metadata and text can be extracted from a wide variety of different file formats. Tika runs a parsing process on the different file formats to obtain all possible information from them.

In the case of image processing, Tika is integrated with TesseractOCR and it gives Solr all the information obtained from the optical character recognition process.

4 Developing the Digesto System

The Digesto system, the same as other information systems, has a relational database whose structure is related to basic system features:

- Administration of users with different access levels
- Regulation administration:
  - Metadata
    - Different visibility levels
    - Characterization by type or issuer
    - Identification number
    - Relation to other regulations and type of relation
  - Digitized contents of the paper print-out

For the Digesto system, paper-based regulations are digitized using a scanner and stored in PDF format (Portable Document Format). Initially, the process consists in just obtaining a digital representation of the paper-based regulation, and does not involve any type of text recognition.

In particular, the digitized content is not stored directly in the relational database, but in a folder within the file system, and a unique file name is assigned to each file. This helps avoid too large databases that could hinder backup tasks.

As regards the Digesto system, the storage scheme consists of a relational database and a file system, as shown in Figure 1.

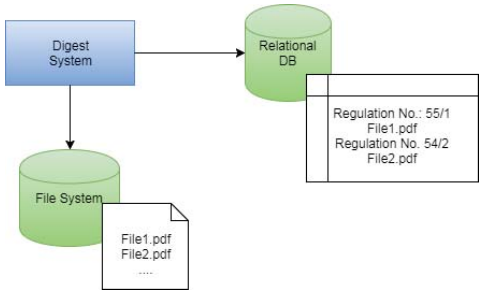

Figure 1. Initial storage scheme.

4.1 Integration of the Digesto System with Solr

The storage scheme used in the Digesto system does not consider the textual contents of the regulations, but rather stores the information in PDF format together with the attributes defined for each of them as a set of metadata.

The content extraction work is managed by Solr and handled by Tika. Using Tika allows recognizing text information within each PDF file attached to each regulation, as well as any images present in the documents. These images can also be analyzed using Tika, so that they are also processed using optical character recognition (OCR) techniques to extract their textual content if possible. OCR tasks are processed by the TesseractOCR library.

The process of generating search indexes and returning search results is handled and run by Solr.

The communication between the Digesto system and Solr is done through HTTP requirements, with information being exchanged in XML or JSON format. Figure 2 depicts the storage scheme integrated with Solr.

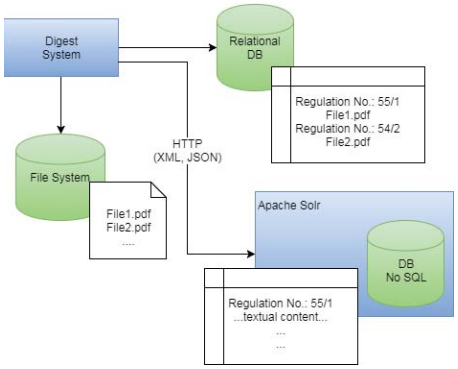

Figure 2. Storage scheme integrated with Solr.

4.2 Synchronization Between the Digesto System and Solr

Synchronizing the information included in a regulation between the Digesto system and Solr is important to guarantee that the search results returned by the system are up to date.

When a new regulation is loaded to the Digesto system, it is stored in the relational database and sent to Solr to be indexed and stored in its own, search-oriented format.

When a regulation is modified, its information has to be totally or partially re-indexed.

Finally, if a regulation is removed, the corresponding information has to be removed from the index as well to prevent the outdated information to be included in future search results.

Even though both databases must be synchronized for the proper operation of the system, the indexed information can be automatically re-built from the relational database of the Digesto system.

4.3 Adding a Regulation to the Digesto System

With the storage infrastructure mounted and configured as shown in Figure 3, the process for adding a regulation to Digesto involves the following steps:

1. Digitization of the paper-based regulation to PDF format using a scanner. (Manual process, external to the system).
2. Loading regulation metadata: date, number, issuer, attached digitized file, etc. (Manual process, in the system).
3. Storing regulation metadata in the relational database. (Automated process).
4. Storing the digitized file in server's file system. (Automated process).
5. Optical Character Recognition of the digitized file to obtain the textual content of the regulation. (Automated process in Solr through Tika and TesseractOCR).
6. Indexing recognized content to allow text searches. (Automated process in Solr).

Figure 4 shows a simplified diagram of the process for adding regulations.

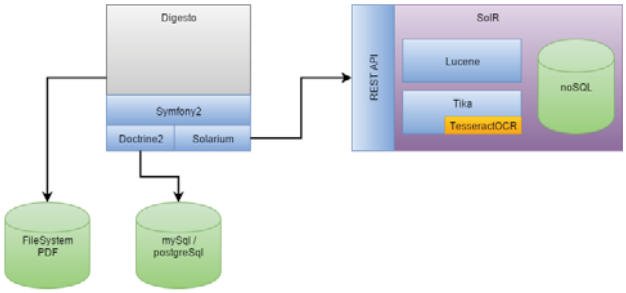

Figure 3. Detailed structure of components in the integrated infrastructure.

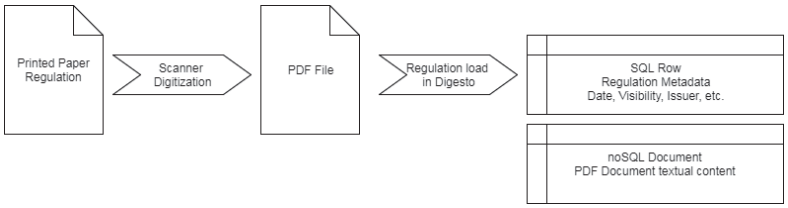

Figure 4. Process for adding regulations

5 Using the Digesto System

The integration of an ad hoc system to perform searches over a set of regulations opens up a number of possibilities that would be too expensive to develop.

The relevant aspects offered by Solr's search engine in the Digesto system are as follows:

Non-SQL storage - Flexibility. The storage method used by Solr is implemented through a non relational database (non-SQL). This type of databases allows indexing entirely heterogeneous documents as regards their structure.

In the case of the Digesto system, each regulation has a number of attributes that are specific to the domain and have to be indexed in the search engine together with the textual information of the regulation.

Some of these attributes should not be treated as text, since they represent dates or numeric values and must allow carrying out comparative searches using mathematical operators such as greater than, less than, not equal to, and so forth. Solr can handle

this situation through the specification of a scheme that defines the type of data that each attribute will contain.

The search engine easily adapts to the specific data structure of the regulations and is tolerant to changes that occur with time due to its non-SQL nature.

Non-SQL storage - Scalability. The non-relational database model presents advantages when it comes to escalating in terms of performance through the use of computer clusters for distributed query processing [9]. Solr can work with clusters, so it can provide mechanisms to improve response speed with large volumes of information.

Query Language. Through Lucene, Solr provides a powerful query language that is easy to understand for the end user. It has many similarities with the searches that can be done with the most popular engines nowadays.

Through Solr, Digesto allows searching for regulations that meet complex conditions, while for the person using Digesto, finding regulations is as easy as finding a page on the Internet.

Plug-In Expansible Architecture. Solr provides an expansible architecture that allows modifying or adding new information indexing methods.

The Digesto system uses this advantage to improve national identification number indexation so that searches can be done either using thousand separators or omitting them.

This is a specific requirement from one of the institutions that is currently using the system, and it was solved through the development of a specific plug-in.

Deferred Indexing. Since the Digesto system and the search engine are separate, when users add regulations to the system, they do not have to wait until the OCR+indexation process ends; they can keep working while the necessary tasks to ready the regulation for inclusion on search results are run in the background.

6 Study cases

The development of the Digesto system was originally aimed at addressing the requirements of the School of Computer Science of the National University of La Plata (UNLP). It went into production during the second half of 2015.

The second application case for the system was at the School of Economic Sciences of the UNLP, where specific changes were implemented as required by the institution and an automated import and indexation feature to add digitized regulations was implemented. The system went live in the first half of 2016.

Both instances of the system, Digesto Info (School of Computer Science) and Digesto Econo (School of Economic Sciences), are currently running on stable versions and available for everyday use at both institutions. Figure 5 shows a

screenshot of the main screen for searching regulations, which is common to all instances.

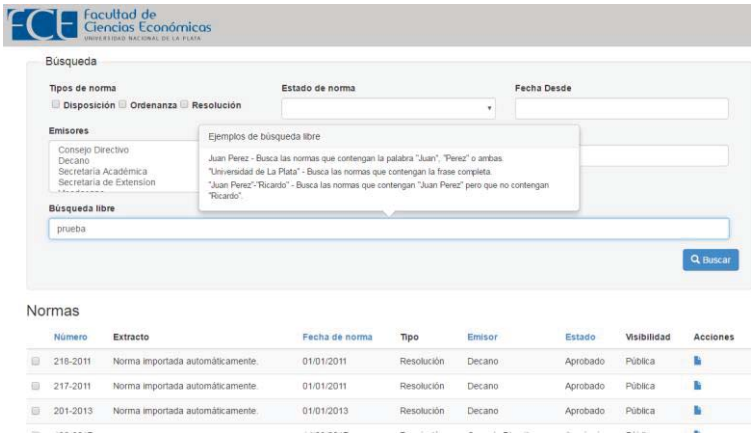

Figure 5. Screenshot of the regulations search screen, Digesto Econo.

7 Results obtained

Both the School of Computer Science and the School of Computer Sciences are currently using the system. The Digesto Econo instance currently holds more than 7200 loaded regulations that can be queried.

Response times for searches on regulation contents are comparable to those of any search process done at popular Internet portals such as Google, Yahoo! or Bing, and they meet the expectations of the users working with the system.

Finding a specific regulation or a set of regulations in relation to specific content is as simple and familiar as searching for a website in a web browser. More advanced queries respond to the same syntax used by Google, and results are presented sorted by relevance.

8 Conclusions

A study was carried out aimed at finding a feasible alternative for the development of a system with requirements that are strongly related to image recognition and agile information search.

The tools selected during this study were analyzed and the Digesto system was developed, which streamlines tasks related to loading and parsing historic and current regulations.

The use of a specific tool such as Solr to carry out the searches provides an added value to the Digesto system that would have been very difficult to achieve if a new tool had been developed from scratch.

Even though the configuration, synchronization and adaptation of Solr is not a trivial process, and specific knowledge of the subject area on which it is going to be used is required, the time invested doing this is negligible compared to the benefits obtained.

The Digesto system that was developed, as well as each of its instances (Info and Econo) meet the requirements and expectations of the end users, since they have a familiar and user-friendly tool that not only makes regulation querying easy, it also simplifies regulation digitization.

9 Future Lines of Work

Since the generic nature of Solr can be applied to different subject areas, in the future we are considering analyzing, configuring and optimizing the tool to improve both metadata extraction and the relevance of the search results returned by Digesto.

Additionally, the regulation digitization process will be improved through the development of a mechanism that will make digital information to remain available after the digitization of paper-based regulations to allow the automated extraction and load of metadata with no user interaction. This would allow users to only carry out the digitization process during the initial load stage; indexation in the Digesto system would be automated.

References

1. Shalin A. Chopra, Amit A. Ghadge: Optical Character Recognition. IJARCCE, Vol. 3, January 2014.
2. Pablo Thomas, Rodolfo Bertone: Introducción a las bases de datos. Prentice Hall - Pearson Education, 2011, ISBN: 9789876153515
3. Tesseract OCR, https://github.com/tesseract-ocr/tesseract [Accessed 21/07]
4. Apache Solr, http://lucene.apache.org/solr/ [Accesed 21/07]
5. Apache Lucene, http://lucene.apache.org/ [Accessed 21/07]
6. Leonard Richardson, Sam Ruby. RESTful Web Services. O'Reilly Media; 1 edition (May 18, 2007), ISBN: 9780596529260
7. Pramod J. Sadalage, Martin Fowler: NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. Addison-Wesley Professional; 1 edition (August 18, 2012). ISBN: 9780321826626
8. Apache Tika, https://tika.apache.org/ [Accessed 21/07]
9. Rajkumar Buyya: High Performance Cluster Computing: Architectures and Systems. Prentice Hall; 1 edition (May 31, 1999). ISBN: 9780130137845